Measuring Uncertainty in Graph Cut Solutions – Efficiently Computing Min-marginal Energies Using Dynamic Graph Cuts

Conference Paper *in* Lecture Notes in Computer Science · July 2006

DOI: 10.1007/11744047_3 · Source: DBLP

2 authors:

Pushmeet Kohli
Microsoft

Philip Hilaire Torr
University of Oxford

Measuring Uncertainty in Graph Cut Solutions - Efficiently Computing Min-marginal Energies using Dynamic Graph Cuts

Pushmeet Kohli Philip H.S. Torr

Department of Computing,
Oxford Brookes University, Oxford.
{pushmeet.kohli, philiptorr}@brookes.ac.uk

Abstract. In recent years the use of graph-cuts has become quite popular in computer vision. However, researchers have repeatedly asked the question whether it might be possible to compute a measure of uncertainty associated with the graph-cut solutions. In this paper we answer this particular question by showing how the min-marginals associated with the label assignments in a MRF can be efficiently computed using a new algorithm based on dynamic graph cuts. We start by reporting the discovery of a novel relationship between the min-marginal energy corresponding to a latent variable label assignment, and the flow potentials of the node representing that variable in the graph used in the energy minimization procedure. We then proceed to show how the min-marginal energy can be computed by minimizing a *projection* of the energy function defined by the MRF. We propose a fast and novel algorithm based on dynamic graph cuts to efficiently minimize these energy projections. The min-marginal energies obtained by our proposed algorithm are exact, as opposed to the ones obtained from other inference algorithms like loopy belief propagation and generalized belief propagation. We conclude by showing how min-marginals can be used to compute a confidence measure for label assignments in labelling problems such as image segmentation.

1 Introduction

Researchers in computer vision have extensively used graph cuts to compute the maximum a posteriori (MAP) solutions for various discrete pixel labelling problems such as image restoration, segmentation and stereo. Graph cuts are preferred over other inference algorithms like Loopy Belief Propagation (LBP), Generalized Belief Propagation (GBP) and the recently introduced Tree Re-weighted message passing (TRW) [1, 2] primarily because of their ability to find globally optimal solutions for an important class of energy functions (sub-modular) in polynomial time [3]. Even in problems where they do not guarantee globally optimal solutions, they can be used to find solutions which are strong local minima of the energy [4]. These solutions for certain problems have been shown to be better than the ones obtained by other methods [5].

[*]This work was supported by the EPSRC research grant GR/T21790/01(P) and the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778.

Graph cuts however do suffer from a big disadvantage. Unlike other inference algorithms, they do not provide any uncertainty measure associated with the solution they produce. This is a serious drawback since researchers do not have any information regarding the probability of a particular latent variable assignment in a graph cut solution. Inference algorithms like LBP, GBP, and TRW provide the user with marginal or min-marginal energies associated with each latent variable. However, these algorithms are not exact. Note that for tree-structured graphs, the simple max-product belief propagation algorithm gives the exact max-marginal probabilities/min-marginal energies[1] for different label assignments in $O(nl^2)$ time where n is the number of latent variables, and l is the number of labels a latent variable can take.

This paper addresses the problem of efficiently computing the min-marginals associated with the label assignments of any latent variable in a Markov Random Field (MRF). Our method can work on all MRFs that can be solved using graph cuts. First, we show how in the case of binary variables, the min-marginals associated with the labellings of a latent variable are related to the *flow-potentials* (defined in section 3) of the node representing that latent variable in the graph constructed in the energy minimization procedure. The exact min-marginal energies can be found by computing these *flow-potentials*. We then show how flow potential computation is equivalent to minimizing *projections* of the original energy function[2].

Minimizing a *projection* of an energy function is a computationally expensive operation and requires a graph cut to be computed. In order to obtain the min-marginals corresponding to all label assignments of all random variables, we need to compute a graph cut $O(nl)$ number of times. In this paper, we present an algorithm based on dynamic graph cuts [6] which solves these $O(nl)$ graph cuts extremely quickly. Our experiments show that the running time of this algorithm i.e. the time taken for it to compute the min-marginals corresponding to all latent variable label assignments is of the same order of magnitude as the time taken to compute a single graph cut.

Overview of Dynamic Graph Cuts Dynamic computation is a paradigm that prescribes solving a problem by dynamically updating the solution of the previous problem instance. Its hope is to be more efficient than a computation of the solution from scratch after every change in the problem. A considerable speedup in computation time can be achieved by this procedure especially when the problem is large scale and changes are few. Dynamic algorithms are not new to computer vision. They have been extensively used in computational geometry for problems such as range searching, intersections, point location, convex hull, proximity and many others [7].

Boykov and Jolly [8] were the first to use a *partially* dynamic st-mincut algorithm in a vision application, by proposing a technique with which they could update capacities of *certain* graph edges, and recompute the st-mincut dynamically. They used this

[1] We will explain the relation between max-marginal probabilities and min-marginal energies later in section 2. To make our notation consistent with recent work in graph cuts, we formulate the problem in terms of min-marginal energies (subsequently referred to as simply min marginals).

[2] A projection of the function $f(x_1, x_2, ..., x_n)$ can be obtained by fixing the values of some of the variables in the function $f(.)$. For instance $f'(x_2, ..., x_n) = f(0, x_2, ..., x_n)$ is a projection of the function $f(.)$.

method for performing interactive image segmentation, where the user could improve segmentation results by giving additional segmentation cues (seeds) in an online fashion. However, their scheme was restrictive and did not allow for general changes in the graph. In one of our earlier papers, we proposed a new algorithm overcoming this restriction [6], which is faster and allows for arbitrary changes in the graph. The running time of this new algorithm has been empirically shown to increase linearly with the number of edge weights changed in the graph. In this paper, we will use this algorithm to compute the exact min-marginals efficiently. To summarize, the key contributions of this paper include:

- A novel relationship between min-marginal energies and node flow-potentials in the residual graph obtained after the graph cut computation.
- A method to compute min-marginals by minimizing energy function projections.
- An extremely fast algorithm based on dynamic graph cuts for efficiently minimizing these energy projections.
- A method to obtain confidence maps for different assignments in labelling problems such as image segmentation.

Organization of the Paper A brief outline of the paper is given next. We discuss MRFs and min-marginal energies in section 2. In section 3, we formulate the st-mincut problem, define terms that would be used in the paper, and describe how certain energy functions can be minimized using graph cuts. In section 4, we show how min-marginals can be found by minimizing projections of the original energy function. We then propose a novel algorithm based on dynamic graph cuts to efficiently compute the minima of these energy projections. In section 5, we show some experimental results of our algorithm.

2 Notation and Preliminaries

We will now describe the notation used in the paper. We will formulate our problem in terms of a *pairwise* MRF[3]. Note that the pairwise assumption does not affect the generality of our formulation since any MRF involving higher order interaction terms can be converted to a *pairwise* MRF by addition of auxiliary variables in the MRF [9].

Consider a random field consisting of a set of discrete random variables $\{x_1, \ldots, x_n\}$ defined on the set $\mathcal{V}$, such that each variable x_v takes values from the label set $\mathcal{X}_v$. We represent the set of all variables $x_v, \forall v \in \mathcal{V}$ by the vector $\mathbf{x}$ which takes values from the set $\mathcal{X}$ defined as $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_n$. Unless noted otherwise, we use symbols u and v to denote values in $\mathcal{V}$, and i and j to denote particular values in $\mathcal{X}_u$ and $\mathcal{X}_v$ respectively. Further, we use $\mathcal{N}_v$ to denote the set consisting of indices of all variables which are neighbours of the random variable x_v in the graphical model. The random field is said to be a MRF with respect to a neighborhood $\mathcal{N} = \{\mathcal{N}_v | v \in \mathcal{V}\}$ if and only if it satisfies the positivity property $\Pr(\mathbf{x}) > 0 \ \forall \mathbf{x} \in \mathcal{X}$, and the Markovian property

$$\Pr(x_v | \{x_u : u \in \mathcal{V} - \{v\}\}) = \Pr(x_v | \{x_u : u \in \mathcal{N}_v\}) \qquad \forall v \in \mathcal{V}. \tag{1}$$

[3] Pairwise MRFs have cliques of size at most two.

The MAP-MRF estimation problem can be formulated as an energy minimization problem where the energy corresponding to the configuration $\mathbf{x}$ is the negative log likelihood of the joint posterior probability of a MRF configuration and is defined as

$$E(\mathbf{x}|\theta) = -\log \Pr(\mathbf{x}|\mathbf{D}) - \text{const.} \tag{2}$$

Here θ is the energy parameter vector defining the MRF [1]. The energy of a configuration for such a pairwise MRF can be written in terms of unary and pairwise energy terms as:

$$E(\mathbf{x}|\theta) = \sum_{v \in \mathcal{V}} \left(\phi(\mathbf{x}_v) + \sum_{u \in \mathcal{N}_v} \phi(\mathbf{x}_u, \mathbf{x}_v) \right) + \text{const.} \tag{3}$$

In the paper, $\psi(\theta)$ is used to denote the value of the energy of the MAP configuration of the MRF and is defined as:

$$\psi(\theta) = \min_{\mathbf{x} \in \mathcal{X}} E(\mathbf{x}|\theta). \tag{4}$$

The term *optimal solution* will be used to refer to the MAP solution in the paper.

Min-marginal energies A min-marginal is a function that provides information about the minimum values of the energy E under different constraints. Following the notation of [1], we define the min-marginal energies $\psi_{v;j}, \psi_{uv;ij}$ as:

$$\psi_{v;j}(\theta) = \min_{\mathbf{x} \in \mathcal{X}, x_v = j} E(\mathbf{x}|\theta), \quad \text{and} \;\; \psi_{uv;ij}(\theta) = \min_{\mathbf{x} \in \mathcal{X}, x_u = i, x_v = j} E(\mathbf{x}|\theta). \tag{5}$$

In words, given an energy function E whose value depends on the variables $(x_1, \ldots, x_n)$, $\psi_{v;j}(\theta)$ represents the minimum energy value obtained if we fix the value of variable x_v to j and minimize over all remaining variables. Similarly, $\psi_{uv;ij}(\theta)$ represents the value of the minimum energy in the case when the values of variables x_u and x_v are fixed to i and j respectively.

2.1 Computing the likelihood of a label assignment

Now we show how min-marginals can be used to compute a confidence measure for a particular latent variable label assignment. Given the function $\Pr(\mathbf{x}|\mathbf{D})$, which specifies the probability of a configuration of the MRF, the max-marginal $\mu_{v;j}$ gives us the value of the maximum probability over all possible configurations of the MRF in which $x_v = j$. Formally, it is defined as:

$$\mu_{v;j} = \max_{\mathbf{x} \in \mathcal{X}; x_v = j} \Pr(\mathbf{x}|\mathbf{D}) \tag{6}$$

Inference algorithms like max-product belief propagation produce the max-marginals along with the MAP solution. These max-marginals can be used to obtain a confidence measure σ for any latent variable labelling as:

$$\sigma_{v;j} = \frac{\max_{\mathbf{x} \in \mathcal{X}, x_v = j} \Pr(\mathbf{x}|\mathbf{D})}{\sum_{k \in \mathcal{X}_v} \max_{\mathbf{x} \in \mathcal{X}, x_v = k} \Pr(\mathbf{x}|\mathbf{D})} = \frac{\mu_{v;j}}{\sum_{k \in \mathcal{X}_v} \mu_{v;k}} \tag{7}$$

where $\sigma_{v;j}$ is the confidence for the latent variable x_v taking label j. This is the ratio of the max-marginal corresponding to the label assignment $x_v = j$ to the sum of the max-marginals for all possible label assignments.

We now proceed to show how these max-marginals can be obtained from the min-marginal energies computed by our algorithm. Substituting the value of $\Pr(\mathbf{x}|\mathbf{D})$ from equation (2) in equation (6), we get $\mu_{v;j} = \max_{\mathbf{x}\in\mathcal{X};x_v=j}\ (\exp\left(-E(\mathbf{x}|\theta) - \text{const}\right))$ or $\mu_{i;j} = \frac{1}{Z}\exp\left(-\min_{\mathbf{x}\in\mathcal{X};x_v=j}\ E(\mathbf{x}|\theta)\right)$, where Z is the partition function. Combining this with equation (5a), we get $\mu_{i;j} = \frac{1}{Z}\exp\left(-\psi_{v;j}(\theta)\right)$. As an example consider a binary label object-background image segmentation problem, where there are two possible labels i.e. object ('ob') and background ('bg'). The confidence measure $\sigma_{v;ob}$ associated with the pixel v being labelled as object can be computed as:

$$\sigma_{v;ob} = \frac{\mu_{v;ob}}{\mu_{v;ob} + \mu_{v;bg}} = \frac{\frac{1}{Z}\exp\left(-\psi_{v;ob}(\theta)\right)}{\frac{1}{Z}\exp\left(-\psi_{v;ob}(\theta)\right) + \frac{1}{Z}\exp\left(-\psi_{v;bg}(\theta)\right)}, \tag{8}$$

$$\text{or} \quad \sigma_{v;ob} = \frac{\exp\left(-\psi_{v;ob}(\theta)\right)}{\exp\left(-\psi_{v;ob}(\theta)\right) + \exp\left(-\psi_{v;bg}(\theta)\right)} \tag{9}$$

Note that the Z's cancel and thus we can compute the confidence measure from the min-marginal energies alone without knowledge of the partition function.

2.2 Computing the M most probable configurations

Another important use of min-marginals has been to find the M most probable configurations (or labellings) for latent variables in a Bayesian network [10]. Dawid [11] showed how min-marginals on junction trees can be computed, which was later used by [12] to find the M most probable configurations of a probabilistic graphical network. Note that the method of [11] is guaranteed to run in polynomial time for tree-structured networks. However, for arbitrary graphs, its worst case complexity is exponential in the number of the nodes in the graphical model.

3 The st-Minimum Cut Problem

In this section we will give a brief overview of graph cuts and show how they can used to minimize energy functions such as the one defined in equation (3). A cut is a partition of the node set V of a graph G into two parts S and $\overline{S} = V - S$, and is defined by the set of edges (i,j) such that $i \in S$ and $j \in \overline{S}$. The cost of a cut $(S,\overline{S})$ is equal to: $C(S,\overline{S}) = \sum_{(i,j)\in E; i\in S; j\in\overline{S}}(c_{ij})$ where c_{ij} is the cost associated with the edge (i,j). For a weighted graph $G(V,E)$ with two special nodes, namely the source s and the sink t, collectively referred to as the terminals, the st-mincut problem is that of finding a cut with the smallest cost satisfying the properties $s \in S$ and $t \in \overline{S}$.

By the Ford-Fulkerson theorem [13], the st-mincut problem is equivalent to computing the maximum flow from the source to the sink with the capacity of each edge equal to c_{ij}. Specifically, while passing flow through the network, a number of edges become saturated. When the maximum amount of flow is being passed in the network, there remains no path from the source to the sink that does not have a saturated edge. In effect, these saturated edges separate the source from the sink and thus by the Ford-Fulkerson theorem, constitute the minimum cut.

Computing the Maximum Flow The Max-flow problem for a capacitated network $G(V, E)$ with a non-negative capacity c_{ij} associated with each edge is that of finding the maximum flow f from the source node s to the sink node t subject to the edge capacity and flow balance constraints:

$$0 \leq f_{ij} \leq c_{ij} \quad \forall (i,j) \in E, \quad \text{and} \tag{10}$$

$$\sum_{i \in N(v)} (f_{vi} - f_{iv}) = 0 \quad \forall v \in V - \{s, t\} \tag{11}$$

where f_{ij} is the flow from node i to node j, and $N(v)$ is the neighbourhood of v.

Residual Graphs, Augmenting Paths and Flow Potentials Given a flow f_{ij}, the residual capacity r_{ij} of an edge $(i, j) \in E$ is the maximum additional flow that can be sent from node i to node j using the edges (i, j) and (j, i). The residual capacity r_{ij} has two components: the unused capacity of the edge (i, j): $c_{ij} - f_{ij}$ and the current flow f_{ji} from node j to node i which can be reduced to increase the flow from i to j. A residual graph $G(f)$ of a graph G consists of the node set V and the edges with positive residual capacity (with respect to the flow f). The topology of $G(f)$ is identical to G. $G(f)$ differs only in the capacity of its edges and so for zero flow i.e. $f_{ij} = 0 \, \forall (i, j) \in E$, $G(f)$ is same as G.

An augmenting path is a path from the source to the sink along unsaturated edges of the residual graph. Augmenting path based algorithms for solving the max-flow problem work by repeatedly finding augmenting paths in the residual graph and saturating them. When no more augmenting paths can be found i.e. the source and sink are disconnected in the residual graph, the maximum flow is obtained.

We define the *flow potentials* of a graph node as the maximum amount of flow that can be pumped between it and the two terminals without invalidating the flow balance (11) and edge capacity (10) constraints of the weighted graph. For a node i, we refer the maximum amount of flow that can be pumped from it is as the source flow potential f_i^s and that into it as the sink flow potential f_i^t. The computation of flow potential is not a trivial process and in essence requires a graph cut to be computed as shown in figure 2. The flow potentials of a particular graph node are shown in figure $1(a)$. Note that in a residual graph $G(f_{\max})$ where $f_{\max}$ is the maximum flow, all nodes on the sink side of the st-mincut are disconnected from the source and thus have the source flow potential equal to zero. Similarly, all nodes belonging to the source have the sink flow potential equal to zero. We will show later that the flow-potentials we have just defined are intimately linked to the min-marginals.

3.1 Minimizing Energies using Graph Cuts

The basic procedure for energy minimization using graph cuts comprises of building a graph in which each cut defines a configuration $\mathbf{x}$, and the cost of the cut is equal to the energy value associated with $\mathbf{x}$ i.e. $E(\mathbf{x}|\theta)$. Kolmogorov and Zabih [3] showed under what conditions energies like (3) can be minimized exactly using st-mincuts. They also described how to construct the graph for this particular class of energy functions. Their work dealt with energy functions involving binary random variables. The conditions and graph construction corresponding to the multiple label case was later given in [14].

The basic graph construction for the minimization procedure works by decomposing the energy function into unary and pairwise energy terms. The MRF energy (3) can be written as:

$$E(\mathbf{x}|\theta) = \theta_{\text{const}} + \sum_{v \in V, i \in \mathcal{X}_v} \theta_{v;i} \delta_i(x_v) + \sum_{(s,t) \in E, (j,k) \in (\mathcal{X}_s, \mathcal{X}_t)} \theta_{st;jk} \delta_j(x_s) \delta_k(x_t), \quad (12)$$

where $\theta_{v;i}$ is the penalty for assigning label i to latent variable x_v, $\theta_{st;ij}$ is the penalty for assigning labels i and j to the latent variables x_s and x_t, and each $\delta_j(x_s)$ is an indicator function which is defined as:

$$\delta_j(x_s) = \begin{cases} 1 \text{ if } x_s = j, \text{ where} j \in \mathcal{X}_s \\ 0 \text{ otherwise} \end{cases},$$

These individual energy terms are represented by weighted edges in the graph. Multiple edges between the same nodes are merged into a single edge by adding their weights. Finally, the st-mincut is found in this graph, which provides us with the MAP solution. The cost of this cut corresponds to the energy of the MAP solution. The labelling of a latent variable depends on the terminal it is disconnected from by the minimum cut. If the node is disconnected from the source, we assign it the value zero and one otherwise. The graph construction for a two node MRF is shown in figure $1(b)$.

4 Computing Min-marginals using Graph Cuts

We will now explain how min-marginal energies can be computed using graph cuts. The total flow f_{total} flowing from the source s to the sink t in a graph is equal to the difference between the total amount of flow coming in to a terminal node and that going out i.e.

$$f_{\text{total}} = \sum_{i \in N(s)} (f_{si} - f_{is}) = \sum_{i \in N(t)} (f_{it} - f_{ti}). \quad (13)$$

We know that the cost of the st-mincut in an energy representing graph is equal to the energy of the optimal configuration. From the Ford-Fulkerson theorem, this is also equal to the maximum amount of flow $f_{\max}$ that can be transferred from the source to the sink. Hence from the minimum energy (4) and total flow equation (13) for a graph in which maxflow has been achieved i.e. $f_{\text{total}} = f_{\max}$, we obtain:

$$\psi(\theta) = \min_{\mathbf{x} \in \mathcal{X}} E(\mathbf{x}|\theta) = f_{\max} = \sum_{i \in N(s)} (f_{si} - f_{is}). \quad (14)$$

Note that flow cannot be pushed into the source i.e. $f_{is} = 0, \forall i \in V$. Thus, we get $\psi(\theta) = \sum_{i \in N(s)} f_{si}$. The MAP configuration $\mathbf{x}^*$ of a MRF is the one having the least energy and is defined as $\mathbf{x}^* = \arg\min_{\mathbf{x} \in \mathcal{X}} E(\mathbf{x}|\theta)$. The min-marginals corresponding to the optimal label assignments for the latent variables are equal to the minimum energy i.e.

$$\psi_{v;x_v^*}(\theta) = \min_{\mathbf{x} \in \mathcal{X}, x_v = x_v^*} E(\mathbf{x}|\theta) = \psi(\theta) \quad (15)$$

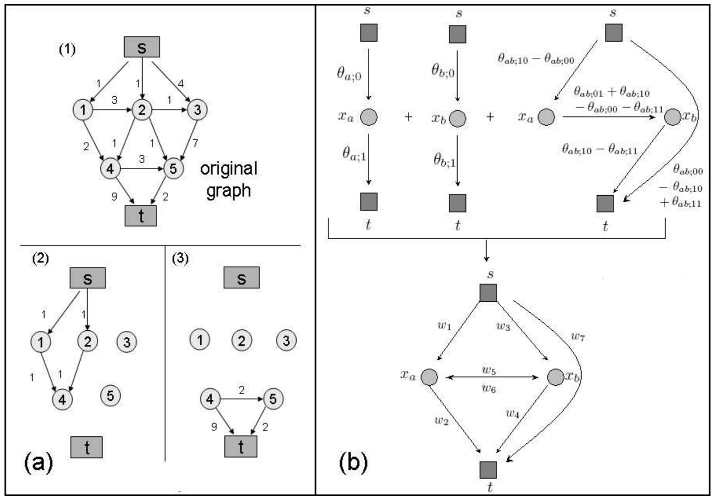

Fig. 1. *a) Illustrating the flow potentials of graph nodes. The figure shows a directed graph having seven nodes, two of which are the terminal nodes, the source s and the sink t. The number associated with each directed edge in this graph is a capacity which tells us the maximum amount of flow that can be passed through it in the direction of the arrow. The flow potentials for node 4 in this graph when no flow is passing through any of the edges are $f_4^s = 2$ and $f_4^t = 11$. b) Energy minimization using graph cuts. The figure shows how individual unary and pairwise terms of an energy function taking two binary variables are represented and combined in the graph. The cost of a st-cut in the final graph is equal to the energy $E(\mathbf{x})$ of the configuration $\mathbf{x}$ the cut induces. The minimum cost st-cut induces the least energy configuration $\mathbf{x}$ for the energy function.*

where x_v^* is the label given to the latent variable x_v in the MAP configuration $\mathbf{x}^*$. Thus the maximum flow equals the min-marginals for the case when the latent variables take their respective MAP labels. The min-marginal energy $\psi_{v;x_v^-}(\theta)$ corresponding to a non-optimal label x_v^- can be computed by finding the minimum value of the energy function projection $E^{'}$ obtained by constraining the value of x_v to x_v^- as:

$$\psi_{v;x_v^-}(\theta) = \min_{\mathbf{x}\in\mathcal{X}, x_v = x_v^-} E(\mathbf{x}|\theta) = \min_{(\mathbf{x}-x_v)\in(\mathcal{X}-\mathcal{X}_v)} E(x_1, .., x_v^-, x_{v+1}..x_n|\theta). \quad (16)$$

In the next paragraph, we will show that this constraint can be enforced in the original graph construction used for minimizing $E(x|\theta)$ by modifying certain edge weights which make sure that the latent variable x_v takes the label x_v^-. The exact modifications needed in the graph for the binary label case are given first while those required in the graph for the multi-label case are discussed later.

Min-marginals and Flow potentials We now show how in the case of binary variables, flow-potentials in the residual graph $G(f_{\max})$ are related to the min-marginal energy values. We will use a and b to represent the MAP and non-MAP label respectively.

Theorem 1. *The min-marginal energy of a binary latent variable x_v is equal to the sum of the max-flow and the flow-potential of the node representing it in the residual*

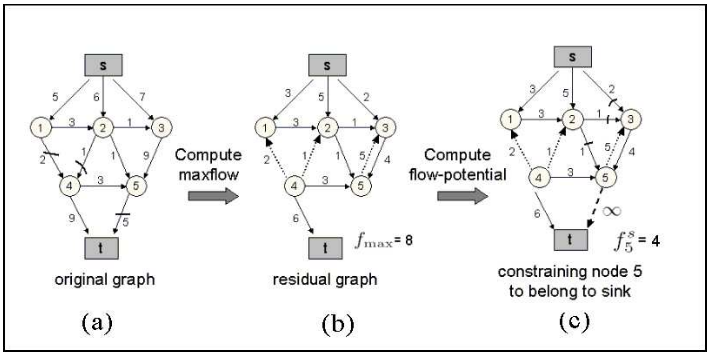

Fig. 2. *Computing min-marginals using graph cuts. In (a) we see the graph representing the original energy function. This is used to compute the minimum value of the energy $\psi(\theta)$ which is equal to the max-flow $f_{\max} = 8$. The residual graph obtained after the computation of max-flow is shown in (b). In (c) we show how the flow-potential f_5^s can be computed in the residual graph by adding an infinite capacity edge between it and the sink and computing the max-flow again. The addition of this new edge constrains node 5 to belong to sink side of the st-cut. A max-flow computation in the graph (c) yields $f_5^s = 4$. This from theorem 1, we obtain the min-marginal $\psi_{5;c} = 8 + 4 = 12$, where T(c) = source(s).*

graph corresponding to the max-flow solution $G(f_{\max})$ i.e.

$$\psi_{v;j}(\theta) = \min_{x \in \mathcal{X}, x_v = j} E(\mathbf{x}|\theta) = \psi(\theta) + f_v^{T(j)} = f_{\max} + f_v^{T(j)} \tag{17}$$

where $T(j)$ is the terminal corresponding to the label j, and $f_{\max}$ is the value of the maximum flow in the graph G representing the energy function $E(\mathbf{x}|\theta)$.

Proof The proof is trivial for the case where the latent variable takes the optimal label. We already know that the value of the min-marginal $\psi_{v;a}(\theta)$ is equal to the lowest energy $\psi(\theta)$. Further, the flow potential of the node for the terminal corresponding to the label assignment is zero since the node is disconnected from the terminal $T(a)$ by the minimum cut[4].

We already know from (16) that the min-marginal $\psi_{v;b}(\theta)$ corresponding to the non-optimal label b can be computed by finding the minimum value of the function E under the constraint $x_v = b$. This constraint can be enforced in our original graph (used for minimizing $E(x|\theta)$) by adding an edge with infinite weight between the graph node and the terminal corresponding to the label a, and then computing the st-mincut on this updated graph[5]. It can be easily seen that the additional amount of flow that would now flow from the source to the sink is equal to the flow potential $f_v^{T(b)}$ of the node. Thus the

[4] The amount of flow that can be transferred from the node to the terminal $T(a)$ in the residual graph is zero since otherwise it would contradict our assumption that the max-flow solution has been achieved.

[5] Adding an infinite weight edge between the node and the terminal $T(a)$ is equivalent to putting a hard constraint on the variable x_v to have the label b. Please note that the addition of an

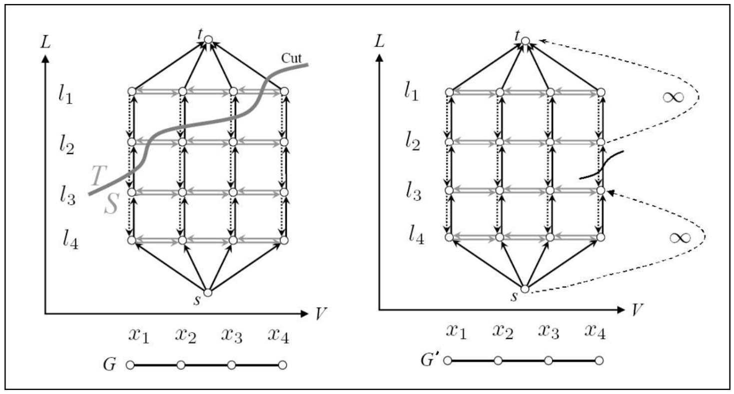

Fig. 3. *Graph construction for projections of energy functions involving multiple labels. The first graph G shows the graph construction proposed by Ishikawa [14] for minimizing energy functions representing* MRF*s involving latent variables which can take more than 2 labels. All the label sets $\mathcal{X}_v \;\forall v \in V$, consist of 4 labels namely l_1, l_2, l_3 and l_4. The* MAP *configuration of the* MRF *induced by the st-mincut is found by observing which data edges are cut (data edges are depicted as black arrows). Four of them are in the cut here (as seen in graph G), representing the assignments $x_1 = l_3$, $x_2 = l_2$, $x_3 = l_2$, and $x_4 = l_1$. The graph G' representing the projection $E' = E(x_1, x_2, x_3, l_3)$ can be obtained by inserting infinite capacity edges from the source and the sink to the tail and head node respectively of the edge representing the label l_3 for latent variable x_4.*

value of the max-flow now becomes equal to $\psi(\theta) + f_v^{T(b)}$ where $T(b)$ is the terminal corresponding to the label b. The whole process is shown graphically in figure 2.

We have shown how minimizing an energy function with constraints on the value of a latent variable, is equivalent to computing the flow potentials of a node in the residual graph $G(f_{\max})$. Note that a similar procedure can be used to compute the min-marginal $\psi_{uv;ij}(\theta)$ by taking the projection and enforcing hard constraints on pairs of latent variables.

Extension to Multiple labels Graph cuts can also be used to optimize certain specific energy functions which involve variables taking multiple labels [14]. Graphs representing the projections of such energy functions can be obtained by incorporating hard constraints in a fashion analogous to the one used for binary variables. In the graph construction for multiple labels proposed by Ishikawa [14], the label of a discrete latent variable is found by observing which data edge is cut. The value of a variable can be constrained or 'fixed' in this graph construction by making sure that the data edge corresponding to the particular label is cut. This can be realized by adding edges of infinite capacity from the source and the sink to the tail and head node of the edge respectively

infinite weight edge can be realized by using an edge whose weight is more than the sum of all other edges incident on the node. This condition would make sure that the edge is not saturated during the max-flow computation as shown in figure 3. The cost of the st-mincut in this modified graph will give the exact value of min-marginal energy associated with that particular labelling.

4.1 Minimizing Energy Function Projections using Dynamic Graph Cuts

Having shown how min-marginals can be computed using graph cuts, we now explain how this can be done efficiently. As explained in the proof of theorem 1, we can compute min-marginals by minimizing projections of the energy function. It might be thought that such a process is extremely computationally expensive as a graph cut has to be computed for each min-marginal computation. While modifying the graph in order to minimize the projection $E^{'}$ of the energy function, we observed that only a few edge weights have to be changed in the original graph[6] as seen in figure 2, where only one infinite capacity edge had to inserted in the graph. In our earlier work [6], we had showed that the st-mincut can be recomputed rapidly for such minimal changes in the problem by using the dynamic graph cut algorithm. The dynamic graph cut algorithm works by updating the residual graph obtained from the previous minimization procedure to reflect the changes in the problem. It then recomputes the st-mincut on this updated residual graph. This scheme enables extremely fast computation of the st-mincut when the number of changes in the problem are few. Our proposed algorithm is given in Table 1.

1. Construct graph G for minimizing the MRF energy E.
2. Compute the maximum s-t flow in the graph. This induces the residual graph G_r consisting of unsaturated edges.
3. If a label assignment is included in the MAP solution obtained in step 2, then the corresponding min-marginal is equal to the energy of the MAP solution.
4. For computing each remaining min-marginal, perform the following operations:
   (a) Obtain the energy projection E' corresponding to the latent variable assignment.
   (b) Construct the graph G' to minimize E'.
   (c) Use dynamic updates as given in [6] to make G_r consistent with G', thus obtaining the new graph $G_r^{'}$.
   (d) Compute the min-marginal by minimizing E' using the dynamic st-mincut algorithm [6] on $G_r^{'}$.

Table 1. *Algorithm for computing min-marginal energies using dynamic graph cuts.*

4.2 Algorithmic Complexity and Experimental Evaluation

We now discuss issues related to the complexity of the algorithm shown in Table 1. Note that in step (4d) of the algorithm, the amount of flow computed is equal to the difference in the min-marginal $\psi_{v;j}(\theta)$ of the particular label assignment and the minimum energy $\psi(\theta)$. Let $\mathcal{Q}$ be the set of all label assignments whose corresponding min-marginals have to be computed. Then the number of augmenting paths to be found during the whole algorithm is bounded from above by: $U = \psi(\theta) + \sum_{q \in \mathcal{Q}} (\psi_q(\theta) - \psi(\theta))$. For the

[6] The exact number of edge weights that have to be changed is of the order of the number of variables whose value is being fixed for obtaining the projection.

case of binary random variables, assuming that we want to compute all latent variable min-marginals i.e. $\mathcal{Q} = \{(u; i) : u \in V, i \in \mathcal{X}_v\}$ and $q_{max} = \max_{q \in \mathcal{Q}}(\psi_q(\theta) - \psi(\theta))$, the complexity of the above algorithm becomes O($(\psi(\theta) + nq_{max})T(n, m)$), where $T(n, m)$ is the complexity of finding an augmenting path in the graph with n nodes and m edges and pushing flow through it. Although the worst case complexity $T(n, m)$ of the augmentation operation is O(m), we observe experimentally that using the dual search tree algorithm of [5], we can get a much better amortized time performance. The average time taken by our algorithm for computing the min-marginals in random MRFs of different sizes is shown in Table 2.

MRF size	10^5	2×10^5	4×10^5	8×10^5
4-neighbourhood	0.18, 0.70	0.46, 1.34	0.92, 3.156	2.17, 8.21
8-neighbourhood	0.40, 1.53	1.39, 3.59	2.42, 8.50	5.12, 15.61

Table 2. *Times (in seconds) taken for min-marginal computation for binary random variables. For a sequence of randomly generated MRFs of a particular size and neighbourhood system, a pair of times is given in each cell of the table. On the left is the average time taken to compute the MAP solution using a single graph cut while on the right is the average time taken to compute the min-marginals corresponding to all latent variable label assignments.*

5 Applications of Min-marginals

Min-marginal energies have been used for a number of different purposes. One of the most important of these has been to compute the M most probable configurations of a MRF [10]. Prior to this work, the use of min-marginals was severely restricted because they were computationally expensive to compute for MRFs having a large number of latent variables. However, our new algorithm is able to handle a MRF of far larger size which opens up possibilities for many new applications. For instance, in the experiments shown in figure 4, the time taken for all min-marginal computations for a MRF consisting of 2×10^5 binary latent variables was 1.2 seconds which is roughly four times the time taken for a single graph cut. Next, we show how min-marginals can be used to obtain a confidence value for any pixel label assignment in the image segmentation problem.

Min-marginals as a confidence measure We have shown in section 2.1 how min-marginals can be used to compute a confidence measure for any latent variable assignment in a MRF. Figure 4 shows the confidence values obtained for a MRF used for modeling the two label (foreground and background) image-segmentation problem as defined in [8]. Note that ideally we would like the confidence map to be black and white showing extremely 'low' or 'high' confidence for a particular label assignment. However, as can be seen from the result, the confidence map contains regions of different shades of grey. Such confidence maps can be used to direct user interaction in the context of interactive image segmentation. In order to remove the ambiguity in the solution, the user could give additional cues in the grey regions.

Recently, a number of image segmentation method have been proposed which couple MRFs with prior information about the shape of the object being segmented. In a

Fig. 4. *Image segmentation with max-marginal probabilities. The first image is a frame of the movie Run Lola Run. The second shows the binary foreground-background segmentation where the aim was to segment out the human. The third and fourth images shows the confidence values obtained by our algorithm for assigning pixels to be foreground and background respectively. In the image, the max-marginal probability is represented in terms of decreasing intensity of the pixel. Our algorithm took 1.2 seconds for computing the max-marginal probabilities for each latent variable label assignment. The time taken to compute the* MAP *solution was 0.3 seconds.*

separate work within this volume [15], we describe how a shape prior generated using an articulated human model can be integrated with the MRF used to solve the image segmentation problem. The effect of incorporating a shape prior on the confidence values of the pixels can be seen in figure 5. Our analysis of uncertainty shows that the incorporation of the shape prior in the image segmentation problem gives better results, and reduces the ambiguity in the solution.

6 Conclusions

In this paper we addressed the long-standing problem of computing the exact min-marginals for graphs with arbitrary topology in polynomial time. We propose a novel algorithm based on dynamic graph cuts [6] that computes the min-marginals extremely efficiently. Our algorithm makes it feasible to compute exact min-marginals for MRFs with large number of latent variables. This opens up many new applications for min-marginals which were not feasible earlier. We have presented one such application in the form of obtaining confidence values for pixel label assignments in the image segmentation problem.

References

1. Kolmogorov, V.: Convergent tree-reweighted message passing for energy minimization. In: AISTATS05. (2005) 182–189

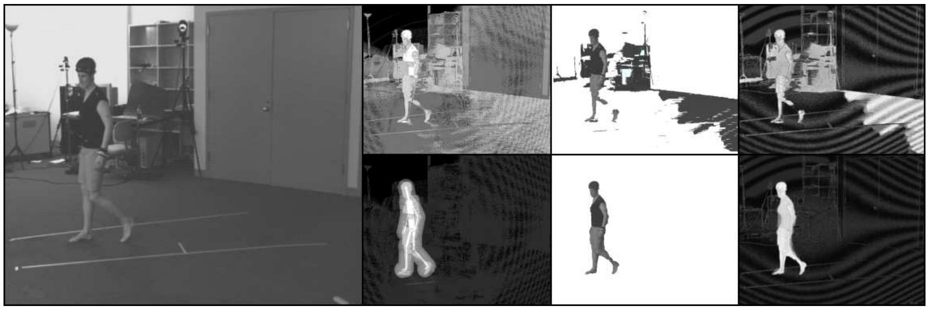

Fig. 5. *Effect of incorporating a shape prior on the confidence values. The first column shows the original image from which we intend to segment out the human. The images in the first row are the result of using only colour information for the segmentation problem. The images in the second row correspond to using a shape prior along with the colour information. In the second column, we see the images representing the difference of the unary penalties $\theta_{v;bg} - \theta_{v;fg}$ for every pixel v. The* MAP *segmentation is shown in the third column, while the images in the fourth column show the confidence values obtained by our algorithm for labelling pixels as foreground.*

2. M. J. Wainwright, T.S.J., Willsky, A.S.: Map estimation via agreement on (hyper)trees: Message-passing and linear-programming approaches. Technical Report UCB/CSD-03-1269 (2003)
3. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts? In: ECCV02. (2002) III: 65 ff.
4. Boykov, Y., Veksler, O., Zabih, R.: Markov random fields with efficient approximations. In: CVPR98. (1998) 648–655
5. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. PAMI **26** (2004) 1124–1137
6. Kohli, P., Torr, P.: Eficiently solving dynamic markov random fields using graph cuts. In: ICCV05. (2005)
7. Chiang, Y.J., Tamassia, R.: Dynamic algorithms in computational geometry. Technical Report CS-91-24 (1991)
8. Boykov, Y., Jolly, M.: Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In: ICCV01. (2001) I: 105–112
9. Weiss, Y., Freeman, W.T.: On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs. IEEE Transactions on Information Theory (2001)
10. Yanover, C., Weiss, Y.: Finding the m most probable configurations in arbitrary graphical models. In: Advances in Neural Information Processing Systems 16. MIT Press (2004)
11. Dawid, P.: Applications of a general propagation algorithm for probabilistic expert systems. Statistics and Computing. **2** (1992) 25–36
12. Nilsson, D.: An efficient algorithm for finding the m most probable configurations in bayesian networks. Statistics and Computing **8** (1998) 159–173
13. Ford, L., Fulkerson, D.: Flows in Networks. Princeton University Press, Princeton (1962)
14. Ishikawa, H.: Exact optimization for markov random fields with convex priors. PAMI **25** (2003) 1333–1336
15. Bray, M., Kohli, P., Torr, P.: Posecut: Simulataneous segmentation and 3d pose estimation of humans using dynamic graph cuts. In: ECCV06. (2006)